Empirical Evaluation of a Service Analysis and Design Methodology

Axel Korthaus, Thomas Kohlborn, Michael Rosemann
Business Process Management
Faculty of Science and Technology
Queensland University of Technology
Brisbane, Australia
Email: {axel.korthaus|t.kohlborn|m.rosemann}@qut.edu.au

James Couzens
Solution Architect
Suncorp
Brisbane, Australia
Email: james.couzens@suncorp.com.au

Abstract

Service-orientation has gained widespread acceptance and is increasingly being employed as a paradigm for structuring both business and IT architectures. An earlier study of extant service analysis and design methodologies discovered a need for holistic approaches that equally account for both business and software services, which motivated the design of a new, consolidated service analysis and design methodology. A challenge in design-oriented research is to evaluate the utility of the newly created artefacts (here: the methodology), as they are often intended to become part of complex socio-technical systems. Therefore, after presenting a brief overview of the consolidated methodology, the paper discusses possible approaches for the "evaluate" phase of this design-science research process and presents the results of an empirical evaluation conducted in an Action Research study at one of Australia's largest financial services providers.

Keywords

Service analysis and design, SOA, methodology, Design Science, evaluation.

INTRODUCTION

The proliferation of the paradigm of service-orientation on both the business and software level consequently leads to a demand for innovative service engineering methodologies that cover both service domains and provide an integrated, holistic approach to ensure business/IT alignment and agility. The literature about current Service-oriented Architecture (SOA) research roadmaps, e.g. Kontogiannis et al. (2007) and Papazoglou et al. (2007), clearly characterises service engineering as being a significant research challenge.

In the context of an Australian ARC Linkage project titled "Service Ecosystems Management for Collaborative Process Improvement" (ARC Linkage Grant: LP0669244), we are currently investigating the foundations of emerging service-oriented business networks. Service Analysis & Design (SAD) is one of the core topics in that project. A central goal in our research is to identify an existing or develop a new, derived methodology for service analysis that accounts for a comprehensive view of service-orientation addressing both business and software services. Since there are many different definitions of service analysis in the literature (cf. Marks and Bell (2006)), it is helpful to explicate our view, which regards service analysis as a phase where the concept of service-orientation is applied to analyse the capabilities provided by an organisation and to identify services that are currently supported and could be supported by IT. This notion includes the analysis of the impact of service-orientation on the business level as well as on the technical level.

Driven by this research goal, our first step was the compilation of a contemporary overview and comparison of existing major service analysis-related service engineering methods covering business and technical viewpoints in order to analyse their characteristics in detail and to evaluate to which extent these approaches are ready for the emerging requirements as stated above (Kohlborn et al. 2009b). This work provided a general overview of the methods that can be chosen to start a SOA endeavour. Additionally, it also led to the conjecture that there is currently a lack of holistic approaches that take both business and software services equally into account. This finding motivated the specification of a new, consolidated service analysis methodology based on a Design Science Research setting (see next section) (Kohlborn et al. 2009a).

Design-oriented research essentially creates innovative artefacts (Hevner et al. 2004), such as the service analysis methodology in our case. An important challenge in this type of research is the actual evaluation of the utility of

the produced artefacts, as they often become part of a complex socio-technical system with many variables that are hard to control. Thus, the *aim of this paper* is twofold. First, we like to report on our endeavour to empirically evaluate the previously published service analysis and design methodology and will focus on an Action Research study as one part of a larger program evaluating (and further developing) the SAD methodology. Second and beyond this narrow goal, we like to contribute to the discussion of Design Science Research (DSR) by elaborating on the experiences of conducting actual artefact assessments in real world settings.

The remainder of this paper is structured as follows. We will first present our research design and a brief overview of the consolidated service analysis methodology. Next, the results of the empirical evaluation of the methodology conducted in an Action Research study with Suncorp, a large Australian company from the financial services sector, are reported and discussed. The paper concludes by summarising the contributions made, the limitations of the presented research and outlining next steps in this research project.

RESEARCH DESIGN

The research problem sketched in the introduction is essentially a Design Science problem. DSR, e.g. as outlined by Hevner et al. (2004), invents or creates new or improved means including constructs, models, methods, and instantiations (March and Smith 1995) to address relevant problems (Venable 2006) and extend the boundaries of human and organizational capabilities (Hevner et al. 2004). It increasingly becomes an accepted mode of research within the IS discipline (Purao et al. 2008), supplementing traditionally more prevalent positivist, interpretivist and critical research paradigms (Venable 2006). According to this approach, IS research is concerned with two design processes, i.e.

- to 'build' purposeful artefacts to address heretofore unsolved problems, and
- to 'evaluate' these artefacts with respect to the utility provided in solving those problems (Hevner et al. 2004; March and Smith 1995).

Hence, as opposed to behavioural science, Design Science aims at providing utility and relevance to practice by innovatively designing an artefact that meets an existing business need or "problem" (Hevner et al. 2004).

When referring to the central task of building a new artefact, Venable (2006) refers to "Solution Technology Invention" as the core of DSR. According to him, solution technology is "any approach to making an improvement in an organisation, including information systems, information technology, systems development methods, algorithms, managerial practices, and many other technologies or techniques" and "each new solution technology is based on or related to other solution technologies that have already been invented, possibly combining them or making small enhancement" (Venable 2006). He proposes a DSR framework (see Figure 1) that can be related to that of Hevner et al., but distinguishes between *naturalistic evaluation* and *artificial evaluation* and explicitly includes *theory building* both before and after the core design research.

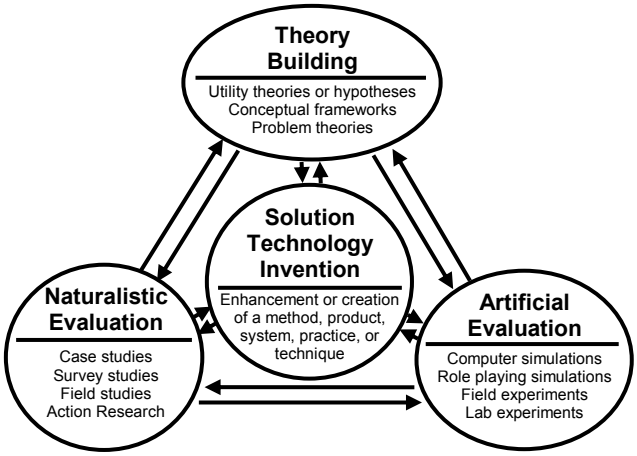

Figure 1: Framework and Context for Design Research (Venable 2006)

The *Theory Building phase* initially requires formulating a utility theory or hypothesising a solution technology, such as "*Solution technology X (when applied properly to problems of type Y) is more effective, efficacious, or efficient than solution technology Z.*" (Venable 2006)

The development and elaboration of the hypothesised solution technology is then the main task in the *Solution Technology Invention* phase. This can include, for example, the development and testing of software, of notations for diagrams or the description of the elements and activities of a new method.

Subsequently, evaluation of the newly developed solution technology is needed to assess its efficiency, efficacy, and/or effectiveness for solving or alleviating these problem(s). According to Venable (2006), "A solution technology is also commonly evaluated in terms of its cost, organisational practicality, and other criteria, relative to other potential means (solution technologies) to solve or alleviate the same problems." The findings from this *Evaluation* phase should then feed back into theory building to confirm or disconfirm existing utility theories. During the *Evaluation* phase, typically empirical methods from the social or natural sciences are employed. A detailed discussion of potential approaches for this phase will be presented after the next section.

Within the framework of DSR as a "research orientation" that can employ different "research methods" (Iivari and Venable 2009), Action Research (AR) is a method that can be very useful, particularly for (but not limited to) the evaluation of the innovative solutions created by the DSR. For a sharp reflection of the relation between AR and DSR refer to (Iivari and Venable 2009) who provide an insightful dissection of the similarities and differences between the two conceptualisations. While Iivari and Venable (2009) view the approaches as compatible, they also warn that due to the fact that when building innovative, cutting-edge artefacts, possible failure is always present, the combination of AR and DSR may be difficult and bear potential risks, particularly to the clients. Generally, AR has a dual goal of contributing to research and practice at the same time (Iivari and Venable 2009) and can help create business knowledge that is both relevant and rigorous (Andriessen 2006), but, as opposed to DSR, it is targeted at a situated goal to be achieved with a specific client in a joint collaboration between researchers and the client (Iivari and Venable 2009). AR allows research ideas to be tested in real world settings, facilitates knowledge transfers between research and practice and allows research ideas to be refined via an iterative learning process (Moody 2005).

For our research on service analysis and design methodologies, we have chosen a research design that utilises the Design Science approach and applies the AR method in order to contribute to theory *and* to deliver practical value to our industry partners. The requirements for a holistic service analysis methodology were derived from observable, emerging trends and needs in practice, and a systematic study of existing approaches confirmed an existing research gap against this "business problem". Based on this identified gap, we hypothesised that a new, consolidated SAD methodology that combines the strengths of complementary approaches and adds new elements would be suitable to address this business problem. The development of the new, consolidated service analysis methodology using method engineering and drawing from related approaches as a foundation represents the actual design of a new artefact in the "build" phase of DSR. As the constructive research method, we used conceptual development, which, according to Iivari (1991) "refers to the development of various models and frameworks which do not describe any existing reality but rather help to create a new one, and which do not necessarily have any 'physical' realization (e.g. IS development 'methodologies')."

THE CONSOLIDATED SERVICE ANALYSIS AND DESIGN METHODOLOGY

The need or desire of an organisation to be transformed according to the service paradigm requires a high-level service analysis and design at organisational level, followed by a full and more detailed analysis and design at the specific service level. Service Analysis captures all activities required to identify and contextualise a service. It can be driven by market requirements (e.g., what services could be profitable offerings?) and/or by various internal artefacts (e.g., strategy maps, process models, data models, application diagrams). The latter case is the core of Service Analysis and this task is focused on the translation of one view (e.g., a process view) into a service-centred view.

The foundation for the central design task of this research project, the development of a structured, consolidated business and software service identification and analysis methodology, was the comprehensive analysis and comparison of 30 extant approaches (Kohlborn et al. 2009b). The objective for the composition of the consolidated methodology was to combine the relative strengths of the extant approaches and to suggest enhancements and extensions based on identified shortcomings. The documentation of the new methodology (Kohlborn et al. 2009a) points out how software services can be identified to support business services to achieve close business and IT alignment. Additionally, the presented approach provides an organisation with a methodology to understand and document its existing capabilities from a service perspective.

The consolidated methodology provides a comprehensive procedural model for service analysis that is subdivided into two main parts, each with different sub-phases. The first part covers the *identification and analysis of business services* by detailing, adapting, and consolidating existing service analysis approaches that focus on the business domain of an organisation. This part has been structured into four distinct phases (Preparation, Identification, Detailing and Prioritisation), each comprising a specific set of activities that may use the outputs of previous phases as inputs. The second part of the consolidated approach describes how *software services can be identified and analysed* that support business services in order to achieve close business and IT alignment. Similar to the first part of the consolidated approach, this part has been structured into distinct phases (Preparation, Identification and Detailing), each comprising specific activities.

Although the overall methodology describes a comprehensive approach for the identification and analysis of both business and software services, the scope of the AR study described in this paper required focusing on certain parts of the methodology, namely the Preparation and Identification phase for software services, as we will explain later. Thus, a brief overview of these two phases will be given in the remainder of this section.

The aim in the *Preparation* phase is to perform preparatory activities that help leverage the service identification phase as well as possible. Thus, the scope of the software service enablement has to be defined, the application portfolio has to be analysed, the processes underlying the different business services have to be, and finally the suitability of processes and applications to be service-enabled has to e analysed as well. The inputs that will be needed in the preparation phase of software services are business services, the SOA strategy and a set of business artefacts, including enterprise architecture models, process models, etc. The preparation phase comprises four main activities, namely "define the scope for service-enablement", "perform application analysis", "perform process decomposition" and "conduct suitability analysis". The outcome of the Preparation phase is a set of business processes that is suitable to be service-enabled and that is decomposed to the most granular process steps with annotated roles and application systems.

The *Identification* phase targets the derivation of software services and comprises the following activities: "identify corresponding entity", "analyse visibility and takeover of process steps", "identify potential service operations", "extract process logic", "define logical context(s)" and "define compositions". Two main sources of information are needed. Firstly, entity models are needed to derive entity services related to the core business objects of each business service. Secondly, detailed process models should be in place to be analysed for service enablement. Furthermore, a designated subset of business services, service design principles and the stakeholders associated with each business service are required as inputs during the Identification phase. The outcome of the identification phase is a set of software service candidates on different hierarchical levels that represent the underlying process of the business service to be supported by software services.

Due to space constraints, we refer the reader to Kohlborn et al. (2009a) for a more detailed description of the different phases and the corresponding activities that constitute the proposed methodology.

RELATED LITERATURE AND EVALUATION FRAMEWORK

Although the consolidated approach combines methodologies that have successfully been applied in practice, its applicability in different contexts of private and public sector organisations needs to be evaluated. Questions that need to be answered include: How can utility/quality/efficacy of the artefact be demonstrated? How can the relative superiority of the artefact be proved? How can both the methodology and the resulting service designs be evaluated, and what does the latter then prove?

The evaluation object in our case is the SAD methodology, which describes a specific approach to a critical stage of service development. The evaluation will be context-dependent, i.e., it is not expected that the methodology will be superior in all circumstances, and evaluations in different organisations might come to different conclusions. We aim at a comparative evaluation that allows a relative assessment of the methodology in relation to alternative approaches. This comparative evaluation should also extend to the output, i.e., the resulting service design model generated through the application of the methodology in a particular case.

In order to gain some insights into shaping an evaluation program for our SAD methodology, we reviewed related literature about evaluation approaches for design artefacts ranging from tools through methods to models. Kitchenham (1996), for example, who describes the DESMET method for evaluating software engineering methods and tools, distinguishes between two main evaluation types (which can also be combined to a hybrid approach):

- *quantitative or objective evaluations* that are directed towards establishing measurable effects of using a method or tool (e.g. reducing production, rework or maintenance time or costs), and
- *qualitative or subjective evaluations* that aim at establishing method or tool appropriateness, i.e. how well a method or tool fits the needs and culture of an organisation (e.g. in terms of required features and characteristics provided by the method/tool).

Moreover, three methods of organising an evaluation are distinguished (Kitchenham 1996):

- formal experiments: here, different subjects (e.g. service developers) are asked to perform tasks using the different methods under investigation;
- case studies: each method under investigation is tried out on a real project;
- surveys: organisations that have used specific methods under investigation on past projects are asked to provide information about that method.

Siau and Rossi (1998) present a review of evaluation approaches for information modelling methods and make a general distinction between *non-empirical* and *empirical techniques*. Their set of non-empirical techniques includes feature comparison, meta-modelling, metrics, paradigmatic analyses, contingency identification, ontological evaluation and approaches based on cognitive psychology, while they list survey, laboratory experiment, field experiment, case study and AR as empirical evaluation techniques.

As mentioned earlier, Venable (2006) distinguishes between *artificial evaluation* and *naturalistic evaluation* as two broad classes of evaluation activities. While the former is predominantly positivist or interpretivist, refers to "evaluating a solution technology in a contrived, non-real way" (Venable 2006) and includes evaluative research methods such as laboratory experiments, field experiments and simulations, the latter may be interpretivist, positivist, and/or critical, "enables a researcher to explore how well or poorly a solution technology works in its real environment – the organisation" (Venable 2006) and builds on evaluation research methods such as case or field studies, surveys, ethnography and AR. Since naturalistic evaluation faces "all of the complexities of human practice in real organisations", Venable (2006) points out that sometimes what can realistically be observed or studied is not a phenomenon itself but rather people's opinions or perceptions. Yet, "successfully solving a problem is often about whether people perceive it to be solved rather than some objectively verifiable phenomenon". A difficulty encountered in naturalistic evaluation is that it may often be impossible to compare with other solution technologies in a real-world setting.

Moody (2005) analyses theoretical and practical issues in evaluating the quality of conceptual models. He also describes possible approaches to developing quality frameworks as foundations for evaluation activities, including theory-based (deductive), experience-based (codification), observation-based (inductive), consensus-based (social) approaches as well as approaches based on synthesis (analytical), derivation (reverse inference) or the Goal-Question-Metric model. In general, publications in the field of method engineering or the Comparative Review of Information Systems (CRIS) series of conferences provide useful input to the interested reader (Siau and Rossi 1998).

The selection of an appropriate (set of) evaluation technique(s) depends on the purpose of the evaluation, the research questions, the opportunities available, the environment, the experience of the researchers and many other factors and requires awareness of the pros and cons of each technique. A detailed discussion would be beyond the scope of this paper, but the interested reader can find a useful and quite comprehensive set of guidelines in Kitchenham (1996).

A major problem that was encountered in the evaluation of the consolidated SAD methodology is caused by its large scope, potentially addressing the whole journey of an organisational transformation towards service-orientation from defining a service strategy through the identification of business services to the derivation of software services from the business processes that support the business services. Due to this large scope, the risk of failure, etc., it is extremely difficult to get access to an industry partner for an all-embracing naturalistic evaluation exercise. We therefore had to define an overall program for the evaluation and further development of the SAD methodology that has been driven by given factors such as accessibility to industry partners and consideration of their particular contexts and interests. However, the goal for the overall program is that eventually all parts of the SAD methodology will be adequately addressed. Table 1 gives an overview of the current status of the evaluation program:

Table 1: Overview of overall evaluation program for the SAD methodology

Evaluated part of SAD methodology	*Project partner*	*SAD methodology applied by*
Whole approach	**Australian Statutory Authority, Perth**	**-**
Business Services	**Australian Government Agency, Brisbane**	**QUT researchers**
Software Services	**Suncorp 1**	**QUT researchers**
Software Services	**Suncorp 2**	**Suncorp architects**

In a first step, the whole approach was presented to and discussed with a team of four Enterprise Architects, managers and consultants of an Australian Statutory Authority in Perth during two two-day workshops. The promising results of the discussions and semi-structured interviews led to the operationalisation of the consolidated SAD methodology in the form of a procedural model, but the methodology was not actually applied. In a second step, a case study at an Australian government agency in Brisbane focussed on the evaluation of the business service-related part of the SAD methodology. The existing procurement process provided by the agency to other agencies was used as the object of study. Based on existing strategic documents and several one hour interviews with officers responsible for the procurement process as input, the QUT researchers applied the *Business Service Identification* and *Detailing* phases of the proposed SAD methodology to describe procurement-related business services and conducted final plausibility checks by feeding back their reports to the government officers for validation.

The third and fourth steps of the evaluation program will be reported on in the remainder of this paper. We chose a multi-perspective approach, aiming at a naturalistic evaluation in a qualitative AR study with an industry partner who faced an urgent need in this area and was interested to set up a trial of our methodology. We selected an AR approach to be able to work with the practitioners in a field setting and derive lessons learned that would be able to serve as a foundation for the improvement of our methodology. To this end, a feature-based evaluation was planned, which required the subjective assessment of the relative importance of different characteristics and how well the proposed methodology ranked along these characteristics in comparison with the industry partner's existing approach. This combination of AR and feature analysis is favourable if the benefits are difficult to quantify, the method user population is limited and the benefits are observable on a single project. The risk, however, is high as the evaluation is based on one person's experience with the method and the evaluation criteria are subjective. Other risks include the influence of human factors that result from sociological effects, such as novelty effects or expectation effects (Kitchenham 1996).

The AR study aimed at empirically evaluating the perceived utility of the developed SAD methodology, whereby the unit of study was the methodology itself on the one hand and the output generated by following the methodology (the resulting service design), on the other hand. Therefore, two sets of criteria or characteristics were required, which will be described in the following.

The *output-related criteria* needed to reflect the outcome of our methodology, i.e. the quality of the service (or SOA) design. In the literature about SOA, various design principles, such as loose coupling and high cohesion, are described that need to be followed to produce "good" service designs (Erl 2007; Papazoglou and van den Heuvel 2006). Based on this literature, we selected coupling, cohesion, reusability, autonomy and completeness as the five criteria to be assessed for the service designs that resulted from the use of the methodologies to be compared. *Coupling* refers to the level of dependency between two or more elements in a service design and should be minimised ("loose coupling"), e.g. because services that are not dependent on other services can be easier maintained and increase the potential for reuse. *Cohesion* refers to the concept of grouping elements based on their functional relatedness to perform a single well-defined task and, as a normative design principle, should be maximised. *Reusability* refers to the applicability of the services in multiple usage scenarios to support different business processes. *Autonomy* denotes the level of independence of a service and the degree of control over its environment, which affects reliability and predictability. With *Completeness*, we refer to the ability of a resulting service design to fully support the corresponding business process and the requirements that result from it.

The *methodology-related criteria* were compiled to assess the proposed methodology as such and the general effects it yields. The seven criteria selected for this purpose include business/IT alignment, efficiency, documentation, usability, innovation, stakeholder focus and completeness. Strong *Business/IT Alignment* is one of the most expected benefits from service-orientation, and we conjecture that particularly the holistic approach of our methodology is suitable to improve the alignment of services with business requirements. As the application of the methodology requires a certain amount of resources, such as time, to produce the desired output, we decided to introduce the criterion *Efficiency*. The *Documentation* criterion should reflect in how far the description of the investigated methodology is a structured step-by-step guideline and facilitates understandability of the approach. *Usability* denotes the ease with which the methodology can be employed in order to achieve the goal of Service Analysis and Design as well as its elegance and clarity. The *Innovation* criterion should capture the investigated methodology's potential to contribute to the innovation of business processes or entire business models. The *Stakeholder Focus* criterion assesses to which extent the methodology at hand takes the requirements of different stakeholders, e.g. human end users, into account. And the *Completeness* criterion in this case finally serves to express in how far the methodology comprises all relevant techniques and activities that are needed to achieve the pursued objective.

For each group, we asked our industry partner to assign a relative weight to each of the criteria in order to indicate the relative importance the respondent attributed to this particular criterion. However, since this set of criteria is not considered to be complete, the respondent was also encouraged to add own criteria that he thought were of importance/interest.

THE SUNCORP ACTION RESEARCH STUDY

In the following, we report on the application of the *Software Service Preparation* and *Identification* phases of our approach at Suncorp, a diversified company in the financial services sector. As one of Australia's leaders in banking, insurance, investment and superannuation focusing on retail customers and small to medium businesses, the Suncorp Group is Australia's sixth largest bank and third largest insurer. With sixteen thousand staff, seven million customers and over eighty billion dollars in assets, it occupies a position in the ASX20. Several mergers, such as the merger with QIDC and Metway Bank to form Suncorp Metway in 1996, the acquisition of GIO and AMP's general insurance business in 2001 and the merger with the Promina Group in 2007, diversified the business and contributed to the group's growth.

We worked with Suncorp's insurance division to apply the above mentioned parts of the methodology in the context of the reorganisation and service-oriented implementation of a "motor claims" and a "home claims" business process. Suncorp had started a Claims Business Model Program some time ago with the intent to identify process improvements that would result in reduced leakage, reduced payments of ineligible claims, and lower handling costs. During these activities it was found that Suncorp's current systems were not sufficiently flexible enough to support the required changes (Couzens 2009). It was then decided that the new claims process should be implemented in a new insurance claims management system from Guidewire Software, the ClaimCenter application (Guidewire 2009). While the initial implementation was in support of personal home claims, implementation projects for claims in worker's compensation, personal motor, commercial motor, commercial property and compulsory third party followed (Couzens 2009).

In the ClaimCenter project, which has been used as a success story of the use of Agile methods (Beck et al. 2001) at Suncorp, integration with a large number of external systems was required and thirteen development teams (including the external vendor and an offshore team) using different development methods had to be coordinated. It was decided that a service-oriented architecture (SOA) approach to integration would be used for customer, policy, payments, general ledger, receipting and claim interfaces with the intent to build them as much as possible for reuse (Couzens 2009).

Against this background, the QUT project team came in and presented the consolidated SAD methodology to a solution architecture team from Suncorp's Business Technology group. Suncorp had found that their approach to SOA and service analysis and design was rather ad-hoc and very much driven by bottom-up integration requirements of their pilot projects, potentially lacking strong alignment of the service designs with the business processes:

"*The current process that we follow tends to be driven by the functional requirements and data requirements of the consumer. This results in a very entity-driven service*[1] *in which the consumer of the service needs to understand a lot more about the state and context of the call that they are making.*" (Suncorp Solution Architect)

An AR study protocol was agreed upon that specified the objectives and the scope of the collaboration as well as the timeframe and the planned deliverables. In a first step, it was decided that the AR study would primarily focus on the identification of software services, and the "motor claims" business process was identified as input to the application of the Software Service Preparation and Identification steps of the SAD methodology developed by QUT. To keep the scope manageable, two sub-processes, namely "claims intake" and "assessment" were selected. The researchers were provided with Suncorp's "motor claims" process models on different levels of hierarchy and additional business artefacts including the SOA Roadmap, the Insurance Domain Model, the ClaimCenter Hub System Architecture Specification v0.1, the ACORD Mapping Guide v1.0 and the UBL Mapping Guide v1.2.2. Based on this input, the researchers applied the Service Preparation and Identification steps as prescribed by the proposed methodology for the two sub-processes and produced two reports that included the resulting service designs.

After the two sub-process "claims intake" and "assessment" had been analysed and software service candidates had been identified through the application of the proposed SAD methodology, the researchers then asked the industry partner to conduct the comparative evaluation of Suncorp's approach ("Method A") versus QUT's methodology ("Method B") and Suncorp's result ("Method A") versus QUT's service design ("Method B") based on the criteria described earlier and the assignment of relative weights. In order to achieve a common understanding of the criteria, the industry partner was emailed the evaluation form together with textual definitions for all criteria to be assessed. A Suncorp solution architect, who acted as the dedicated team leader for related software projects within Suncorp and, due to his professional background, was well grounded in relevant methodological knowledge and experience, conducted the actual evaluation after having consulted with his architecture team. The solution architect's assessment of the SAD methodology developed by QUT was based on the understanding the architect had gained from the presentation of the methodology by QUT researchers in face-to-face meetings and documentation of the methodology given to him in addition. The results of this evaluation are shown in Tables 2 and 3, where the first table refers to the evaluation of methodology-related criteria and the second table reflects output-related criteria. They were sent back by email together with several explanatory comments not shown in the tables. No additional criteria were added by the respondent.

The evaluation results show that QUT's proposed methodology was perceived as being superior to Suncorp's integration requirements-driven ad-hoc approach, particularly due to the increased business/IT alignment achieved through analysing the business process to identify software services.

[1] An entity-driven service is built around business objects, such as customer, order, claim, etc. Thus, such a service typically offers context-agnostic operations that have a high probability to be used in different scenarios. These operations can comprise "CRUD" functionalities to manage the lifecycle of the specific entity.

Tables 2 and 3: Comparative evaluation of Suncorp's ("Method A") and QUT's ("Method B") approach

Table 2: Comparative evaluation of methodology-related criteria			
Criteria (methodology-related)	Relative weight	Method A {1-5} 1=very low/bad 5=very high/good	Method B {1-5} 1=very low/bad 5=very high/good
Business/IT alignment	**25%**	**3**	**4**
Efficiency	**15%**	**2**	**3**
Documentation	**5%**	**2**	**3**
Usability	**25%**	**2**	**3**
Innovation	**10%**	**2**	**4**
Stakeholder focus	**15%**	**3**	**4**
Completeness	**5%**	**1**	**4**
	100%	**2.35**	**3.55**

Table 3: Comparative evaluation of output-related criteria			
Criteria (output-related)	Relative weight	Method A {1-5} 1=very low/bad 5=very high/good	Method B {1-5} 1=very low/bad 5=very high/good
Loose coupling	**25%**	**4**	**2**
Cohesion	**15%**	**0**	**4**
Reusability	**5%**	**4**	**2**
Autonomy	**25%**	**2**	**4**
Completeness	**10%**	**2**	**4**
	100%	**2.60**	**3.00**

Regarding the output-related criteria, the results are mixed. As reflected in the table, the first impression Suncorp got was a lower reusability of the resulting services in the case of QUT's approach. However, it turned out that this impression of poor reusability was due to insufficient process visibility, as the first project was limited to the scope of a single process ("motor claims"). As soon as the number of processes covered was extended in the second project (see below), it became apparent that the services started to belong to the same generic classes and the service design could be re-factored to take the operations up one level to make them more generic. This pointed to the need to do a "shallow end-to-end sweep" across all the different claims processes (motor, home, workers comp. etc.) to get these higher level services before drilling down into the specific area currently dealt with. Another cause is the fact that QUT's approach suggests a fully service-oriented approach that includes the encapsulation of process logic into process and task services, which are more specialised than pure entity services. Suncorp's approach however was limited to entity services, which are naturally loosely coupled, data-oriented and very reusable. Business and application logic in Suncorp's approach is hard-coded in the applications, which is often not the most desirable solution. These circumstances also explain the lower value for the "loose coupling" criterion in QUT's case: since Suncorp only defined entity services, there is no coupling between the services (but to the applications), whereas the further-reaching service design as proposed by QUT's approach also modularises and encapsulates process and application logic in dedicated services, which are then of course coupled with the entity services they invoke but still constitute a more desirable software architecture from a global perspective.

After this first project, a second project was conducted in order to increase the validity of the assessment. This project focussed on the "home claims" business process in the context of a project to build Web and mobile phone front-ends for customers. In this case, two Suncorp solution architects applied the methodology themselves to come up with a service design without support by the researchers for the "claims intake", "claims withdrawal" and "document upload" sub-process of "home claims". The earlier assessment by the lead solution architect as reflected in Tables 2 and 3 was reconfirmed during this hands-on exercise on the part of the industry partner. In parallel, two industry students from university, who were interning at Suncorp at the time, independently developed software services based on the traditional approach used at Suncorp, i.e. bottom-up based on functional requirements. The comparison of the results of these two approaches also turned out to be in favour of the proposed approach.

LESSONS LEARNED

An important lesson learned in the naturalistic evaluation of the proposed methodology so far is that its original structure and description let it appear to be too monolithic and "waterfall-like". It should be possible to use parts of the methodology, even if other parts are not of interest. For example, if an organisation does not yet think in business services, but in business processes only, it should be possible to start at that level to derive software services (as it was done in the case with Suncorp). Therefore, different entry points into the methodology will need to be made explicit to be able to start the process at different stages.

In that context, it will also be necessary to assess the compatibility of the proposed methodology with an overall agile development strategy as followed at Suncorp. It has to be evaluated whether the service analysis and design process prescribed by the proposed methodology violates any agile principles stated in the Agile Manifesto (Beck et al. 2001), which for example values individuals and interactions over processes and tools, working software over comprehensive documentation and responding to change over following a plan.

Another lesson learned referred to the fact that in realistic settings, the methodology must be able to deal with imperfect input, such as partially inaccurate business process models or the lack of a stakeholder model as in the case of Suncorp. Suncorp suggested to build in flexibility or pragmatic guidance into the methodology, e.g. by providing feedback loops from the service identification phase to the business process modelling phase. Suncorp's solution architect expects these feedback loops to help improve the reusability of a service.

Finally, it has become evident that large organisations must be able to tailor the process to suit the way the organisation works. To this end, as mentioned above, we not only aim at modularising the methodology and potentially specify variants that cater for different circumstances, but we also plan to provide tool support for this configurable SAD methodology that allows users to derive a relevant subset of methodology, e.g. based on the particular set of business artefacts that are available as input into the service development process.

CONCLUSION AND OUTLOOK

In this paper, we have presented the results of an empirical evaluation of a new, consolidated service analysis and design methodology that supports the application of the service-oriented paradigm on both the business and the technical level and aims at improving the widely sought-after business/IT alignment in organisations. We have characterised this research as Design Science and showed how AR as a research method can be applied in combination with others. We have discussed potential issues of the naturalistic evaluation of a design artefact (such as a new methodology) and proposed an evaluation framework that can guide the evaluation task. We reported on the case of Suncorp, where parts of our SAD methodology were applied in a real-world scenario, followed by an evaluation of the perceived quality of the methodology and the generated service design. The lessons learned in this use case will inform the improvement of the methodology in a future AR loop.

Limitations of our study include the potentially incomplete literature review of extant service analysis and design methodologies that motivated the design of the new, consolidated methodology, limitations in the design phase of this research, which includes the complete development of the new methodology based on existing approaches, and limitations in the AR phase, such as the lack of a case that allowed to test the methodology as a whole as well as other budgetary and accessibility issues, so that the generalisability of the findings is uncertain. The evaluation also has limitations due to two levels of subjectivity: first in selecting the evaluation criteria by the researchers and second in assessing these criteria by just one single AR study partner.

Due to the fact that Suncorp committed to an agile approach to software development, our next steps with this industry partner will include the assessment and potential modification of our SAD methodology with regard to its compatibility with such an agile software development approach. We are also planning to provide a configurable SAD reference process model and a corresponding configuration questionnaire to allow the tailoring of the reference model to individual needs. In order to provide tool support for this configurable SAD reference process model, we plan to leverage an existing process configuration tool developed at QUT.

REFERENCES

Andriessen, D. 2006. "Combining Design-Based Research and Action Research to Test Management Solutions," in: *The 7th World Congress Action Research*. Groningen, The Netherlands.

Beck, K., Beedle, M., Bennekum, A.v., Cockburn, A., Cunningham, W., Fowler, M., and others. 2001. "Manifesto for Agile Software Development." Retrieved 1. July, 2009, from http://www.agilemanifesto.org/

Couzens, J.A. 2009. "Implementing an Enterprise System at Suncorp Using Agile Development," in: *The 20th Australian Software Engineering Conference (ASWEC 2009)*. Gold Coast, Australia.

Erl, T. 2007. *SOA: Principles of Service Design*. Upper Saddle River, NJ: Prentice Hall.

Guidewire. 2009. "Guidewire Claimcenter - the Flexible Claims Management System for Property & Casualties Insurers." Retrieved 1 July, 2009, from http://www.guidewire.com/our_solutions/claimcenter

Hevner, A.R., March, S.T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp 75-105.

Iivari, J. 1991. "A Paradigmatic Analysis of Contemporary Schools of IS Development," *European Journal of Information Systems* (1:4), pp 249-272.

Iivari, J., and Venable, J. 2009. "Action Research and Design Science Research - Seemingly Similar but Decisively Dissimilar," *17th European Conference on Information Systems*, Verona, Italy.

Kitchenham, B. 1996. "DESMET: A Method for Evaluating Software Engineering Methods and Tools," Technical Report TR96-09, University of Keele, Staffordshire.

Kohlborn, T., Korthaus, A., Chan, T., and Rosemann, M. 2009a. "Identification and Analysis of Business and Software Services - A Consolidated Approach," *IEEE Transactions on Services Computing* (2:1), pp 50-64.

Kohlborn, T., Korthaus, A., Chan, T., and Rosemann, M. 2009b. "Service Analysis - A Critical Assessment of the State of the Art," *17th European Conference on Information Systems*, Verona, Italy.

Kontogiannis, K., Lewis, G.A., Smith, D.B., Litoiu, M., Muller, H., Schuster, S., and Stroulia, E. 2007. "The Landscape of Service-Oriented Systems: A Research Perspective," *SDSOA '07: Proceedings of the International Workshop on Systems Development in SOA Environments*, Washington, DC, USA: IEEE Computer Society.

March, S.T., and Smith, D. 1995. "Design and Natural Science Research on Information Technology," *Decision Support Systems* (15:4), pp 251-266.

Marks, E.A., and Bell, M. 2006. *Service-Oriented Architecture. A Planning and Implementation Guide for Business and Technology*. Hoboken, NJ: John Wiley & Sons.

Moody, D.L. 2005. "Theoretical and Practical Issues in Evaluating the Quality of Conceptual Models: Current State and Future Directions," *Data & Knowledge Engineering* (55), pp 243-276.

Papazoglou, M.P., Traverso, P., Dustdar, S., and Leymann, F. 2007. "Service-Oriented Computing: State of the Art and Research Challenges," *COMPUTER* (40:11), pp 38-45.

Papazoglou, M.P., and van den Heuvel, W.-J. 2006. "Service-Oriented Design and Development Methodology," *International Journal of Web Engineering and Technology (IJWET)* (2:4), pp 412-442.

Purao, S., Baldwin, C.Y., Hevner, A., Storey, V.C., Pries-Heje, J., Smith, B., and Zhu, Y. 2008. "The Sciences of Design: Observations on an Emerging Field," Working Paper 09-056, Harvard Business School.

Siau, K., and Rossi, M. 1998. "Evaluation of Information Modeling Methods - A Review," in: *Thirty-First Hawaii International Conference on System Sciences*. pp. 314-322.

Venable, J. 2006. "A Framework for Design Science Research Activities," in: *The 2006 Information Resource Management Association Conference*. Washington, DC, USA: Idea Group Publishing, Hershey, Pennsylvania, USA.

ACKNOWLEDGEMENTS

Parts of this research have been funded by a research project within the Australian Research Council (ARC) Linkage Schema (grant code LP0669244), including financial support from SAP Research and the Queensland Government.

COPYRIGHT